ELISTdb.8100.Final.SOL7.SAM

Defense Information Infrastructure (DII)
Common Operating Environment (COE)

System Administrator's Manual (SAM)
for the
Enhanced Logistics Intratheater Support Tool (ELIST)
Database Segment Version 8.1.0.0

for Solaris 7

26 February 2002

Prepared for:

Military Traffic Management Command
Transportation Engineering Agency
720 Thimble Shoals Boulevard
Newport News, VA 23606

Prepared by:

Argonne National Laboratory
Decision and Information Sciences Division
9700 South Cass Avenue
Argonne, IL 60439

Table of Contents

List of Tables

1. Scope

This document is the *System Administrator's Manual (SAM) for the Enhanced Logistics Intratheater Support Tool (ELIST) Database Segment*. It covers errors that can arise during the segment's installation and deinstallation, and it outlines appropriate recovery actions. It also tells how to extend the database storage available to Oracle if a datastore becomes filled during the use of ELIST. The latter subject builds on some of the actions that must be performed when installing this segment, as documented in the *Installation Procedures (IP) for the Enhanced Logistics Intratheater Support Tool (ELIST) Global Data Segment, Database Instance Segment, Database Fill Segment, Database Segment, Database Utility Segment, Software Segment, and Reference Data Segment* (referred to in portions of this document as the *ELIST IP*).

The information in this document is expected to be of use only rarely. Other than errors arising from the failure to follow instructions, difficulties are not expected to be encountered during the installation or deinstallation of the segment. The need to extend database storage likewise typically arises infrequently. Most administrators will only need to be aware of the help that is provided in this document and will probably not actually need to read and make use of it.

1.1 Identification

The ELIST Database Segment is one of seven segments that make up the DII COE ELIST mission application.

Table 1 identifies all the segments of the ELIST mission application. In the table, each segment is given a number by which it may be referenced in this document. The table also gives the name, the segment type (and, if a data segment, the segment scope), the current version number, and the directory name assigned to each segment.

Table 1. Segments of the ELIST Mission Application

Segment Number	Segment Name	Segment Type / Scope	Version Number	Directory Name
1	ELIST Global Data Segment	Data / Global	8.1.0.0	`ELISTglob`
2	ELIST Database Instance Segment	Data / Segment	8.1.0.0	`ELISTdbinst`
3	ELIST Database Fill Segment	Data / Local	8.1.0.0	`ELISTdbfill`
4	ELIST Database Segment	Database	8.1.0.0	`ELISTdb`
5	ELIST Database Utility Segment	Software	8.1.0.0	`ELISTdbutil`
6	ELIST Software Segment	Software	8.1.0.0	`ELISTexec`
7	ELIST Reference Data Segment	Data / Local	8.1.0.0	`ELISTrefdata`

All seven segments have the following identification properties in common:

Segment Prefix[1]**:** ELIST

Platform(s)[2]**:** Sun/Solaris 7

DII COE Versions: 4.2.0.0P4 or later

All seven of the ELIST segments must be installed before you can use the ELIST mission application.[3]

Refer to the *Introduction to the Enhanced Logistics Intratheater Support Tool (ELIST) Mission Application and its Segments: Global Data Segment, Database Instance Segment, Database Fill Segment, Database Segment, Database Utility Segment, Software Segment, and Reference Data Segment* for the following:

- an overview of the mission application and all of its segments in the context of the application;
- the definitions of key concepts and terms used throughout the ELIST documentation;
- a complete list of the available ELIST documentation.
- a brief history of ELIST; and
- basic information pertinent to the client/server configuration and installation of the ELIST segments.

[1] Note carefully that all segments have the same prefix. This is not typical of multisegment DII COE mission applications.
[2] Implementation of the ELIST segments for PC/Windows NT 4.0 will follow shortly. This documentation covers only the Sun/Solaris 7 platform but will be supplemented or replaced when an implementation becomes available for NT.
[3]To save space, however, the ELIST Database Fill Segment can be removed after successfully installing the ELIST Database Segment.

2. Referenced Documents

The following other documents are referenced in this document.

2.1 Government Documents

2.1.1 DII COE ELIST Documents

Introduction to the Enhanced Logistics Intratheater Support Tool (ELIST) Mission Application and its Segments: Global Data Segment Version 8.1.0.0, Database Instance Segment Version 8.1.0.0, Database Fill Segment Version 8.1.0.0, Database Segment Version 8.1.0.0, Database Utility Segment Version 8.1.0.0, Software Segment Version 8.1.0.0, and Reference Data Segment Version 8.1.0.0 for Solaris 7, ELIST.8100.Final.SOL7.Intro, Argonne National Laboratory, 26 February 2002

Installation Procedures (IP) for the Enhanced Logistics Intratheater Support Tool (ELIST) Global Data Segment Version 8.1.0.0, Database Instance Segment Version 8.1.0.0, Database Fill Segment Version 8.1.0.0, Database Segment Version 8.1.0.0, Database Utility Segment Version 8.1.0.0, Software Segment Version 8.1.0.0, and Reference Data Segment Version 8.1.0.0 for Solaris 7, ELIST.8100.Final.SOL7.IP, Argonne National Laboratory, 26 February 2002

System Administrator's Manual (SAM) for the Enhanced Logistics Intratheater Support Tool (ELIST) Database Instance Segment Version 8.1.0.0 for Solaris 7, ELISTdbinst.8100.Final.SOL7.SAM, Argonne National Laboratory, 26 February 2002

2.1.2 Other DII COE Documents

System Administrator's Manual (SAM) for Database Administrator Runtime (DBAdmR) Version 3.0.2.0 for Solaris 7, CM Number 43773, FGM Inc., 4 April 2001

Application Program Interface Reference Manual (APIRM) for Database Administrator Server (DBAdmS) Segment Version 3.0.2.0 for Solaris 7, CM Number 43785, FGM Inc., 4 April 2001

2.1.3 Other ELIST Documents

N/A.

2.1.4 Other Government Documents

N/A.

2.2 Non-Government Documents

N/A.

This page intentionally left blank.

3. Installation Overview

Installation and deinstallation of the ELIST Database Segment are documented in the *ELIST IP*.

In contrast to the installation of the ELIST Database Instance Segment, the installation of this segment does not require the administrator to make any decisions. The administrator will be required to enter the password that was chosen and assigned to the database instance during the installation of the ELIST Database Instance Segment, however.

This page intentionally left blank.

4. Error Recovery Guidelines

This section discusses the possible errors that can arise during the installation and deinstallation of the ELIST Database Segment, and it presents guidance that can be helpful in recovering from those errors.

4.1 General Approach to Error Handling in the ELIST Database Segment

4.1.1 General Approach to Error Handling During Installation

The ELIST Database Segment is installed primarily for its side effect on the environment outside the segment, which is the creation of the ELIST database schema consisting of tablespaces and tables (along with the relevant DBO accounts, user roles, and indexes) in the database instance previously secured by the installation of the ELIST Database Instance Segment. In addition, if the installation is a *reinstallation* of the segment following its deinstallation (perhaps to install a new version), and if the ELIST Database Instance Segment has not been deinstalled and reinstalled, it also regrants the relevant roles to the users of the database instance that existed when the segment was deinstalled (because those roles were necessarily dropped at that time).

The principal issue influencing the approach to error handling during the installation of the ELIST Database Segment is the realization that, even if something goes wrong part of the way through the process of creating the database schema, vestiges of that schema are likely to be left around in a partially created state. For example, datastores may have been allocated; database owner (DBO) accounts, user roles, tables and indexes partially created; and DBO accounts left connected. Similarly, Oracle system tables may have been updated. Recovery from this unpredictable error state requires that actions be taken either to make further forward progress towards the ultimate goal of creating the database schema or, alternatively, to reverse the external effects of its partial creation, restoring the state to that which existed prior to its creation. For many reasons, the latter choice is more practical and more likely to be successful. (No amount of "retrying" without other remedial steps is likely to have an effect if, for instance, insufficient storage exists for the schema in the database instance.)

It was reasoned that the best tool for reversing the external effects of a partially created database schema is the segment's own deinstallation script, provided that it is sufficiently robust to cope with unexpected states. Therefore, the general approach to error handling during the installation of the ELIST Database Segment is to continue past the first detected error (after displaying an appropriate message), performing as many subsequent steps of the installation process as make sense, and *to leave the segment installed*.[4] Continuing the installation past the point of the first failure is likely to generate additional messages, calling attention to secondary failures that are direct or indirect consequences of the initial failure. At the conclusion of the installation process, a summary message points out that one or more errors have occurred, that the segment has been left installed, that the installer should consult this document, and that explicitly deinstalling the segment is a suitable way to clean up; it also names the log file created during the installation process, which the installer should consult for additional details about the failure(s), if there is to be any hope of better success on a retry.

[4] This last point is somewhat unorthodox; it is more common for segment installation scripts to signal a failure to the COE Segment Installer, causing it to clean up by removing all vestiges of the partially installed segment. If this were to be done, obviously there would be no deinstallation script to use in an attempt to reverse the external effects of the partially completed database creation.

The error handling strategy that was implemented is as outlined above, with one modification: Some initial checks are performed before anything is changed, and if these checks fail, the segment is left *uninstalled*. This is reasonable, because no recovery is necessary given that no state changes have occurred yet. The error messages that are displayed in this case clearly state what has happened and what is being done.

It might have been reasonable to have the installation script invoke the deinstallation script automatically after a failure is detected, saving the installer from the need to explicitly deinstall the segment. By *not* doing that, however, we give the installer adequate time to investigate the situation before planning a course of action. In some cases, the installer may also be able to determine that it makes more sense to use *ad hoc* methods (such as SQL scripts or other tools to carry out steps that failed, perhaps with some changes) to proceed in a forward direction, rather than to clean up and remove the partially created schema.

Suggestions that may be helpful for recovering from particular errors during installation are presented later in Section 4.2.1.

4.1.2 General Approach to Error Handling During Deinstallation

Deinstallation of the ELIST Database Segment has the side effect of removing the ELIST database schema (*i.e.*, dropping the ELIST database tables, indexes, tablespaces, and owner accounts from the database instance). Though it would be a rare event, removal of the database schema can fail at any of several steps for a variety of reasons. One strategy for dealing with such errors is to leave the segment installed if any are detected, so that presumably the installer can "try again." However, it is reasonable to expect that subsequent attempts to deinstall the segment will not meet with any more success. Thus, this strategy could lead to a situation in which the segment cannot be deinstalled.

Instead, the following approach was taken. Some initial checks are performed, and failure at this stage indeed leaves the segment still installed. Recovery from these failures is easy and reliable, so no problem is posed by this. Following a failure that occurs after these initial checks have succeeded, the deinstallation process continues to perform subsequent steps that make sense, and the segment is left in a deinstalled state. Both the initial failure and any subsequent failures cause appropriate messages to be displayed, with the final summary message clearly indicating whether or not the segment was deinstalled. Note that failure of one or more deinstallation steps may leave vestiges of the database schema in existence, with no means within the segment (which has been deinstalled) to assist in their cleanup. In such cases, an experienced DBA will be required to employ *ad hoc* methods to clean up the schema. In this case, the final summary message names the log file in which additional clues to the nature of the failure can be found, and it directs the installer's attention to the present document.

The deinstallation script is also designed to be robust enough to be used in an attempt to clean up after a failed installation (see the previous section). Although it will usually produce a variety of error messages in such a case, it is likely to succeed in its mission, and no further action need be taken.

Suggestions that may be helpful for recovering from particular errors during deinstallation are presented later in Section 4.2.2.

4.2 Handling of Specific Error Situations

4.2.1 Handling of Specific Errors During Installation

This section discusses specific failures that can be diagnosed by error messages during the installation of the ELIST Database Segment.

Note that the intent is not to display all possible messages here. Those that arise in *normal* processing are adequately covered in the IP and are not repeated here.

Various failures can occur during the initial processing performed by the installation procedure, leaving the segment uninstalled. The handling of these errors is described first.

The installation procedure obtains some of the information it needs from the `/h/data/global/DBAdmS/etc/DBAServerStatus` file. This file is not updated automatically when database instances are created or removed; it must be updated by invoking the **DB Server Snapshot** feature of the **DBAdmS** application (Database Administrator Server segment) following such events. If the administrator neglected to perform the relevant step of the installation or deinstallation instructions in the *ELIST IP* when the ELIST Database Instance segment was last installed, the installation procedure for the ELIST Database Segment may not find the information it needs, and the following message is displayed:

```
A server status file needs to be updated before
installing this segment.  Log on as a user (a DBA) with
access to the features of the Database Administrator Server
(DBAdmS) segment and execute the DB Server Snapshot feature.
Then try installing this segment again.
```

The administrator should follow the directions in the message. The account to which the administrator should log on is the DBA account mentioned in the "Account Preparation" section of the *ELIST IP*.

Normally, the Oracle server associated with the ELIST database instance is running. However, if the installation procedure finds the server not running, it tries to start the server. If that fails, a message of the following form is displayed:

```
DBAdmSStartServer failed to start the server for the
<instance> database instance (it returned a status code
of 1).  Aborting the install.  Consult the System
Administrator's Manual (SAM) for this segment.
```

(In all the message templates shown in this manual, *<instance>* stands for the name of the ELIST database instance, which was chosen or supplied during the installation of the ELIST Database Instance Segment.) The only real remedy available in this situation is to try to start the server manually. To do that, the administrator should log on using the DBA account referenced above and invoke the **Server Control** feature of the **DBAdmR** application. Follow the instructions for that tool in the *System Administrator's Manual (SAM) for Database Administrator Runtime (DBAdmR)*. If that fails to start the server, and the DBA cannot resolve the problem, report it to DISA (see Section 7).

The installation procedure next prompts for the password of the `SYSTEM` user of the database instance. That password was chosen during the installation of the ELIST Database Instance

Segment, unless the instance already existed before that (in which case it was chosen when the instance was created). If the password is incorrectly entered, a message of the following form is displayed:

```
DBAdmSPromptPassword failed to verify the password for the
SYSTEM user of the <instance> database instance that you
entered, or you canceled the password prompt (it returned a
status code of <status>).  Aborting the install.  Consult
the System Administrator's Manual (SAM) for this segment.
```

(Here and elsewhere in this manual, *<status>* is the return code from the API under discussion.) The administrator should try again, being careful to enter the password correctly. Instructions for assigning a new password can be found in the *System Administrator's Manual (SAM) for the Enhanced Logistics Intratheater Support Tool (ELIST) Database Instance Segment*. Note, however, that all of the methods for changing the password described there require one to connect first to the database using a valid DBA account and corresponding password.

Beyond this point, the installation procedure follows the standard sequence of steps for database installation as portrayed in the DII COE I&RTS, with only minimal adaptation. The steps in this process are as follows:

1. **Allocate Storage**. The DBAdmSCreateDS tool of the Database Administrator Server (DBAdmS) segment is used to allocate the datastores.
2. **Create Database Owner**. The DBO account is actually created as a byproduct of the use of DBAdmS in Step 1; it is not explicitly performed as a separate step, and consequently this step is empty.
3. **Create User Roles**. Roles for database user accounts are created in this step.
4. **Create Database**. The schema objects (*i.e.*, tables and indexes) are created.
5. **Load Data**. Initial data provided in the form of `.dmp` files by the ELIST Database Fill Segment is imported into the newly created tables. All the tables are loaded by a single import.
6. **Create Constraints**. Inasmuch as indexes were created in Step 4, and not deferred to this step, this step is empty.
7. **Assign Grants**. (a) Access permissions on the tables are granted to the previously created roles. In the ELIST Database Segment, this step also performs the following nonstandard action, which is not described in the I&RTS: (b) it determines whether the current installation is a *reinstallation* of the segment following a previous deinstallation and, if so, regrants those roles to the database users who had them when the segment was last deinstalled. Step 7b is performed only if the ELIST Database Instance Segment has not *also* been deinstalled and reinstalled between the previous deinstallation of the ELIST Database Segment and its current (re)installation.
8. **Disconnect Owner Accounts**. The `CONNECT` privilege is revoked from the database owner (DBO) accounts, in accord with DII COE rules.

The schema objects of the ELIST database are, for historical reasons, owned by two DBO accounts, `elist` and `etpfdd`. The major adaptation of the installation process, beyond incorporating Step 2 into Step 1 and Step 6 into Step 4, and adding an action to Step 7, is that the entire sequence of eight steps is performed twice, first for the schema objects owned by the `elist` DBO account and then for the schema objects owned by the `etpfdd` DBO account.

If any failure occurs during the processing of these eight steps, first for the `elist` DBO account and then for the `etpfdd` DBO account, processing continues with whatever subsequent steps are relevant, and the segment is left installed. This potential skipping of subsequent steps means that each step, in general, has prerequisite steps that must succeed before it is attempted. Table 2 shows the prerequisites for each step. (By way of example, Step 5 is only attempted if Steps 1 and 4 both succeed. As another example, the table also reveals that failure of Step 1 causes Steps 4, 5, 7a, and 7b to be skipped.) Messages indicate when steps are being skipped due to failures in previous steps.

Step	Prerequisite Steps								
	1	**2**	**3**	**4**	**5**	**6**	**7a**	**7b**	**8**
1									
2									
3									
4	X								
5	X			X					
6									
7a	X		X	X					
7b	X		X	X			X		
8									

Table 2. Prerequisites for Performing Installation Steps

At the end of the installation procedure, one of two summary messages (discussed later) is displayed, indicating whether or not any failures occurred.

The handling of errors detected during this part of the installation procedure (*i.e.*, during these eight steps) is described next. In the following messages, *<dbo>* is either `elist` or `etpfdd`, indicating which DBO's schema objects encountered the error.

If Step 1 fails to allocate the tablespaces and create the DBO account, a message of the following form is displayed:

```
Failed to allocate the <dbo> tablespaces and create the
<dbo> DBO account.  It could be that the DBA neglected to
invoke DBAdmRIdentifyStorage to reserve storage for the
tablespaces before installing this segment.  This install
will continue, but the installed segment will not be
functional.  After the install, consult the System
Administrator's Manual (SAM) for this segment.
```

Without a successful allocation of the tablespaces and creation of the DBO account, few of the subsequent steps are attempted (because they would surely fail), and subsequent messages call attention to the fact that they are being skipped.

The most likely cause of this error is conveyed by the message; in fact, other causes are not imaginable. The *ELIST IP* directs the administrator to invoke `DBAdmRIdentifyStorage` before installing the ELIST Database Segment. If the administrator neglects to do so, `DBAdmSCreateDS` displays a dialog window giving the administrator the option of invoking `DBAdmRIdentifyStorage` interactively before proceeding, but it is possible to decline the invitation. In this situation, the administrator should recover by explicitly deinstalling the ELIST

Database Segment (ignoring error messages), invoking `DBAdmRIdentifyStorage` (following the instructions in the *ELIST IP*), and then reinstalling the segment.

If Step 3 fails to create the applicable user role, a message of the following form is displayed. In the message, *<role>* gives the name of the role (either `elist_user` or `etpfdd_user`).

```
Failed to create the <role> role.  This install will
continue, but the installed segment will not be functional.
After the install, consult the System Administrator's Manual
(SAM) for this segment.
```

It is hard to imagine how this step can fail; if it does, it should probably be reported to DISA. The log should be consulted for any available clues. If the DBA can figure out what went wrong, and how to correct it, the DBA might try to do so, using either the **DBA Studio** or the **SQL Plus** features of the **ORAS** application. (Log in using the DBA account mentioned previously.) It will also be necessary to perform Steps 7a and 7b, which will be skipped as noted in subsequent messages. An experienced administrator can do that, but no real help is provided beyond some comments in the various installation scripts, which can be found at

- `/h/ELISTdb/SegDescrip/PostInstall`
- `/h/ELISTdb/install/Install_DB`
- `/h/ELISTdb/install/ELIST_DB_Defn_Scripts/*`
- `/h/ELISTdb/install/ETPFDD_DB_Defn_Scripts/*`

These should be studied and understood before proceeding. Alternatively, the segment can be deinstalled.

If Step 4 fails to create the schema objects belonging to the applicable DBO, a message of the following form is displayed:

```
Failed to create the <tables> owned by
<dbo>, and their associated constraints and indexes.  This
install will continue, but the installed segment will not
be functional.  After the install, consult the System
Administrator's Manual (SAM) for this segment.
```

In this message, *<tables>* is `elist` if *<dbo>* is `elist`; it is `etpfdd and target` if *<dbo>* is `etpfdd`.

Once again, sensible scenarios explaining the failure of this step are hard to conjure up, and failures should be reported to DISA. Consult the log. It is likely that some tables were successfully created, and others were not. Since all the tables are created by a single SQL script, it is not particularly easy to recover incrementally (*i.e.*, by creating only the missing tables). The segment can safely be deinstalled.

If Step 5 fails to load the tables with the initial data contained in the ELIST Database Fill Segment, a message of the following form is displayed:

```
Failed to import initial data into the <tables>
tables owned by <dbo>.  This install will continue,
but the installed segment will not be functional.
After the install, consult the System Administrator's
Manual (SAM) for this segment.
```

The *<tables>* fragment is as described above. As before, it is hard to imagine how this step can fail. If it does, an experienced DBA might try to perform the import by using the **DBA Studio** feature of the **ORAS** application, when logged on to the DBA account mentioned previously. The segment can alternatively be deinstalled. (However, if Step 5 is the only step that fails, there is no reason to deinstall the segment if the import can be performed manually.)

If Step 7a fails to grant the relevant privileges to the applicable user role, the message is:

```
Failed to assign grants to the <dbo>_user role.  This
install will continue, but the installed segment will not
be functional.  After the install, consult the System
Administrator's Manual (SAM) for this segment.
```

An experienced and adventuresome DBA can try to perform Step 7a manually, following the advice given above for Step 3, or the segment can be deinstalled. It will also be necessary to perform Step 7b manually, since the failure of Step 7a causes Step 7b to be skipped.

If Step 7b fails to regrant the user roles to one or more existing database users, a message of the following form is displayed for each such *<user>*:

```
Failed to regrant the <dbo>_user role to <user>.
```

This is not fatal; if it is the only error reported, there is no reason to deinstall the segment. After studying the log and understanding the reason for the failure, the DBA can use the **DBA Studio** or **SQL Plus** features of the **ORAS** application to regrant the user roles to the relevant users. The lists of users who had the `elist_user` and `etpfdd_user` roles when the ELIST Database Segment was last deinstalled can be found in the `Users_With_The_elist_user_Role` and the `Users_With_The_etpfdd_user_Role` files, respectively, in the `/h/data/global/ELISTglob/data/Users_When_ELISTdb_Deinstalled` directory (provided that the ELIST Database Instance Segment has not been deinstalled and reinstalled between the deinstallation of the ELIST Database Segment and its current reinstallation).

Finally, a message of the following form is displayed if Step 8 fails.

```
Failed to disconnect the <dbo> DBO account.  After the
install, consult the System Administrator's Manual (SAM) for
this segment.
```

This situation is also not fatal, but to maintain security of the database in the ways envisioned in the I&RTS, the DBA should revoke the `CONNECT` privilege from the `elist` and `etpfdd` DBO accounts by using the **DBA Studio** or **SQL Plus** feature of the **ORAS** application, or by arranging to repeat the failed step manually. Note that the message is *not* produced if Step 1 also fails; in that case, failure of Step 8 is considered to be caused by the failure of Step 1 to create

the DBO account. Attempting Step 8 even if Step 1 fails recognizes that Step 1 can conceivably leave the DBO account in existence even if it fails to allocate the datastores.

After all eight steps have been performed for the both of the DBO accounts, one of two messages is displayed, depending on whether any errors are detected during the performance of those steps. If no errors are detected, the following message is displayed:

```
The ELIST database appears to have been successfully
created.  However, you should check the log file,
/h/data/global/ELISTglob/data/Logs/ELISTdb/Install_Log,
carefully to make sure there were no undetected failures.
If you find failures other than "ORA-00942: table or view
does not exist," which is innocuous, you should consult the
System Administrator's Manual (SAM) for this segment.
```

If any such undetected failures are found to be noted in the log, you should determine which step failed and then follow the instructions given above for failures in that step. Note that under certain circumstances, Step 5 can fail in undetected ways (the parameters necessarily used in the Oracle `import` performed in that step result in mismatches between the table definitions and the data being imported being considered as warnings only, causing the install script to be oblivious to such mismatches, if they occur). If this happens in Step 5, it is an indication that internal testing prior to the release of this segment failed to note the error, and DISA should be informed. Step 5 normally should never fail.

If errors are detected, the following message is displayed:

```
One or more failures were detected during the creation of
the ELIST database, as previously noted.  In addition to
carrying out the instructions previously given, you should
check the log file,
/h/data/global/ELISTglob/data/Logs/ELISTdb/Install_Log,
carefully to make sure there were no undetected failures.
If you find failures that were not previously noted, other
than "ORA-00942: table or view does not exist," which is
innocuous, you should consult the System Administrator's
Manual (SAM) for this segment.
```

In this case, the administrator should carry out the instructions previously given, supplemented by the instructions given immediately above regarding undetected failures noted in the log.

4.2.2 Handling of Specific Errors During Deinstallation

This section discusses specific failures that can be diagnosed by error messages during the deinstallation of the ELIST Database Segment.

Note that the intent is not to display all possible messages here. Those that arise in *normal* processing are adequately covered in the IP and are not repeated here.

Various failures can occur during the initial processing performed by the deinstallation procedure, leaving the segment installed. The handling of these errors is described first.

The deinstallation procedure checks to make sure that other ELIST segments that depend on this one have already been deinstalled; if they have not, an error message states that they must be

deinstalled first, and the deinstallation is aborted, leaving this segment installed. In this case, the obvious recovery is to deinstall the dependent segments first, then reattempt to deinstall the ELIST Database Segment.

Next, the deinstallation procedure checks to make sure that the Database Administrator Server (DBAdmS) segment is still installed. (Although "required" by the ELIST Database Segment, it could have been deinstalled prior to the attempt to deinstall the latter segment.) If it is not found to be installed, the following message is displayed:

```
The Database Administrator Server (DBAdmS) segment, which is
required, has been deinstalled, hence the deinstallation
of the ELIST Database Segment cannot proceed.  Reinstall
that segment, then repeat the deinstallation of this
segment.
```

The administrator must do as directed.

The deinstallation procedure obtains some of the information it needs from the `/h/data/global/DBAdmS/etc/DBAServerStatus` file. This file is not updated automatically when database instances are created or removed; it must be updated by invoking the **DB Server Snapshot** feature of the **DBAdmS** application (Database Administrator Server segment) following such events. If the administrator neglected to perform the relevant step of the installation or deinstallation instructions in the *ELIST IP* when the ELIST Database Instance segment was last installed, the deinstallation procedure for the ELIST Database Segment may not find the information it needs, and the following message is displayed:

```
A server status file needs to be updated before
deinstalling this segment.  Log on as a user (a DBA) with
access to the features of the Database Administrator Server
(DBAdmS) segment and execute the DB Server Snapshot feature.
Then try deinstalling this segment again.
```

The administrator should follow the directions in the message. The account to which the administrator should log on is the DBA account mentioned in the "Account Preparation" section of the *ELIST IP*.

Normally, the Oracle server associated with the ELIST database instance is running. However, if the deinstallation procedure finds the server not running, it tries to start the server. If that fails, a message of the following form is displayed:

```
DBAdmSStartServer failed to start the server for the
<instance> database instance (it returned a status code
of 1).  Aborting the deinstall.  Consult the System
Administrator's Manual (SAM) for this segment.
```

The only real remedy available in this situation is to try to start the server manually. To do that, the administrator should log on using the DBA account referenced above and invoke the **Server Control** feature of the **DBAdmR** application. Follow the instructions for that tool in the *System Administrator's Manual (SAM) for Database Administrator Runtime (DBAdmR)*. If that fails to start the server, and the DBA cannot resolve the problem, report it to DISA (see Section 7).

The deinstallation procedure next prompts for the password of the `SYSTEM` user of the database instance. That password was chosen during the installation of the ELIST Database Instance

Segment, unless the instance already existed before that (in which case it was chosen when the instance was created). If the password is incorrectly entered, a message of the following form is displayed:

```
DBAdmSPromptPassword failed to verify the password for the
SYSTEM user of the <instance> database instance that you
entered, or you canceled the password prompt (it returned a
status code of <status>).  Aborting the deinstall.  Consult
the System Administrator's Manual (SAM) for this segment.
```

The administrator should try again, being careful to enter the password correctly. Instructions for assigning a new password can be found in the *System Administrator's Manual (SAM) for the Enhanced Logistics Intratheater Support Tool (ELIST) Database Instance Segment*. Note, however, that all of the methods for changing the password described there require one to connect first to the database using a valid DBA account and corresponding password.

Beyond this point, the deinstallation procedure follows the standard sequence of steps for database installation as portrayed in the DII COE I&RTS, with only minimal adaptation. The steps in this process are as follows:

1. **Remove User Roles**. When the user roles are removed, Oracle necessarily revokes the roles from all the users who currently have them. To support the regranting of those roles to the users in Step 7b of the reinstallation of this segment, if the ELIST Database Instance Segment has not also been deinstalled and reinstalled, the removal of the roles is preceded by a nonstandard action, which is not described in the I&RTS: (a) In storage belonging to the ELIST Global Data Segment, save a list of the users who have each role. Following this, the action described in the I&RTS is performed: (b) Remove the user roles.
2. **Remove Database**. An extra step, not described in the I&RTS, is added here also: (a) Reconnect the database owner (DBO) accounts. This is obviously necessary, because those DBO accounts are needed to perform the step described by the I&RTS: (b) Drop the tables and other schema objects (indexes, *etc*.).
3. **Remove Storage**. The `DBAdmSDropDS` tool of the Database Administrator Server (DBAdmS) segment is used to drop the datastores.
4. **Remove Database Owner**. The DBO account is actually dropped as a byproduct of the use of DBAdmS in Step 3; it is not explicitly performed as a separate step, and consequently this step is empty.
5. **Remove Files**. It is not clear what was intended for this step, since the tablespaces are removed in Step 3. Consequently, this step is empty.

The major adaptation of the deinstallation process, beyond adding an action to Steps 1 and 2, is that the entire sequence of five steps is performed twice, first for the schema objects owned by the `elist` DBO account and then for the schema objects owned by the `etpfdd` DBO account.

If any failure occurs during the processing of these five steps, first for the `elist` DBO account and then for the `etpfdd` DBO account, processing continues with whatever subsequent steps are relevant, and the segment is left deinstalled. This potential skipping of subsequent steps means that each step, in general, has prerequisite steps that must succeed before it is attempted. Table 3 shows the prerequisites for each step. (As the table shows, there is only one prerequisite: if Step 2a fails, Step 2b is skipped.) Messages indicate when steps are being skipped due to failures in previous steps.

Step	Prerequisite Steps						
	1a	1b	2a	2b	3	4	5
1a							
1b							
2a							
2b			X				
3							
4							
5							

Table 3. Prerequisites for Performing Deinstallation Steps

At the end of the installation procedure, one of two summary messages (discussed later) is displayed, indicating whether or not any failures occurred.

The handling of errors detected during this part of the deinstallation procedure (*i.e.*, during these five steps) is described next. In the following messages, *<dbo>* is either `elist` or `etpfdd`, indicating which DBO's schema objects encountered the error.

Step 1a can fail in two different ways. First, a check is made to see whether the *<dbo>*`_user` role was successfully created in Step 3 of the previous *installation* of the segment. If not, that step failed during installation. To avoid mistaking the absence of the *<dbo>*`_user` role for the absence of any users to whom the role has been granted, the list of users who have the *<dbo>*`_user` role is *not* written during Step 1a of the deinstallation, so as to preserve the list that existed when the segment was last installed (prior to the installation that failed). The following message is displayed in this situation:

```
NOT saving a list of users who have the <dbo>_user role,
because that role was not created during the last
installation of this segment.  The previous list of users
will persist and will be used when the problems preventing
the installation of this segment are corrected, and the
segment is properly installed.  After the deinstall,
consult the System Administrator's Manual (SAM) for
this segment.
```

This behavior assumes that the segment is being deinstalled in response to a failure during its installation. Nothing further is required to recover from a failure of this *deinstallation* step.

If the *<dbo>*`_user` role *was* successfully created in Step 3 of the previous installation of the segment, an attempt is made to save the list of users who now have that role. If deinstallation Step 1a fails while saving that list, a message of the following form is displayed:

```
Failed to save a list of users who have the <dbo>_user role.
That role will be dropped anyway, and consequently it will
be revoked from current users.  If the segment is
reinstalled, the DBA wil have to regrant the role to the
ELIST database users manually.  After the deinstall,
consult the System Administrator's Manual (SAM) for this
segment.
```

The dropping of the role (in Step 1b, below) is consistent with the goal of removing as many of the database's artifacts as possible, given that the segment will be positively deinstalled.

When a failure of this second kind occurs, the list of users with the *<dbo>*_user role is emptied to guard against regranting the role to an obsolete list of such users (left over from the *previous* deinstallation). As the message says, the DBA will have to regrant the role to the appropriate users manually after the segment is reinstalled. The difficult part of that is determining to which users the role should be regranted. The DBA should log in using the DBA account mentioned previously and invoke the **SQL Plus** feature of the **ORAS** application. The DBA can then attempt to query the instance to determine which users have the elist_user role and which have the etpfdd_user role, using the following SQL commands:

```
select grantee from dba_role_privs
   where granted_role = 'ELIST_USER';
select grantee from dba_role_privs
   where granted_role = 'ETPFDD_USER';
```

Alternatively, an Admininstrative ELIST User can log in and use the **List User Accounts** feature of the **ELISTdbutil** application (the ELIST Database Utility Segment), which essentially issues the same commands. However, it was commands like these whose failure led to the display of the foregoing message, so it may be assumed that these commands will not work. If they do not, the DBA can query the instance to list all users, using the following SQL command:

```
select username, created from all_users;
```

Note that this will list a fairly large number of system or DBA accounts in addition to the desired user accounts, and the DBA will have to sort through the list to determine which are legitimate user accounts and then which of *those* are likely to have had the the elist_user and etpfdd_user roles. Following that, the DBA should regrant the elist_user and etpfdd_user roles to the appropriate set of users, using either the **DBA Studio** feature or the **SQL Plus** feature of the **ORAS** application. If the latter feature is used, the SQL commands to issue for each *<user>* are

```
grant elist_user to <user>;
grant etpfdd_user to <user>;
```

Returning now to the failures possible during each of the deinstallation steps, if Step 1b fails to remove (drop) the *<dbo>*_user role, a message of the following form is displayed:

```
Failed to drop the <dbo>_user role; continuing.
```

This failure is not fatal. If the ELIST Database Segment is being deinstalled in order to install a new version, the failure can probably be ignored. Note that Step 3 of the subsequent reinstallation of the segment will then fail, but that failure can also be ignored. The failure of Step 1b can also be ignored if the ELIST Database Instance Segment is to be deinstalled, since the role will disappear when the database instance is removed. However, if the administrator wishes to leave everything in as clean a state as possible, the role can be removed by using either the **DBA Studio** feature or the **SQL Plus** feature of the **ORAS** application. If the latter feature is used, the SQL commands to issue are

```
drop role elist_user;
drop role etpfdd_user;
```

If Step 2a fails to reconnect the DBO account, a message of the following form is displayed:

```
Failed to reconnect the <dbo> DBO account; continuing.
```

Although this will cause Step 2b to be skipped, this failure can be ignored, because Step 3 will still be attempted and will likely succeed in removing the tablespaces, making the removal of the tables and other schema objects in Step 2b a moot point; Step 3 will also remove the DBO account, making its reconnection in Step 2a a moot point.

If Step 2b fails to remove the tables owned by the DBO, the following message is displayed:

```
Failed to drop the <tables> owned by
<dbo>, and their associated constraints and indexes;
continuing.
```

In this message, *<tables>* is `elist` if *<dbo>* is `elist`; it is `etpfdd and target` if *<dbo>* is `etpfdd`. As implied above, the failure of Step 2b is not fatal, and it can be ignored because Step 3 will still be attempted and will likely succeed in removing the tablespaces, making the removal of the tables and other schema objects in Step 2b a moot point.

If Step 3 fails to remove the tablespaces, a message of the following form is displayed:

```
Failed to drop the <dbo> tablespaces and the <dbo> DBO
account; continuing.
```

This message can be ignored if the ELIST Database Instance Segment is also to be deinstalled, otherwise the administrator should try to remove the `ELIST_DATA`, `ELIST_INDEX`, `ETPFDD_DATA`, and `ETPFDD_INDEX` tablespaces using the **DBA Studio** feature of the **ORAS** application. Alternatively, in a command window the DBA can invoke the `DBAdmSDropDS` API of the **DBAdmS** application; see the *Application Program Interface Reference Manual (APIRM) for Database Administrator Server (DBAdmS) Segment* for instructions. However, the latter technique is unlikely to work simply because the deinstallation procedure has already tried and failed to use `DBAdmSDropDS`.

After all five steps have been performed for the both of the DBO accounts, one of two messages is displayed, depending on whether any errors are detected during the performance of those steps. If no errors are detected, the following message is displayed:

```
The ELIST database appears to have been successfully
removed.  However, you should check the log file,
/h/data/global/ELISTglob/data/Logs/ELISTdb/Deinstall_Log,
carefully to make sure there were no undetected failures.
If you find failures, you should consult the System
Administrator's Manual (SAM) for this segment.
```

If any such undetected failures are found to be noted in the log, you should determine which step failed and then follow the instructions given above for failures in that step.

If errors are detected, the following message is displayed:

```
One or more failures were detected during the removal of the
ELIST database, as previously noted.  In addition to
carrying out the instructions previously given, you should
check the log file,
/h/data/global/ELISTglob/data/Logs/ELISTdb/Deinstall_Log,
carefully to make sure there were no undetected failures.
If you find failures that were not previously noted, you
should consult the System Administrator's Manual (SAM) for
this segment.
```

In this case, the administrator should carry out the instructions previously given, supplemented by the instructions given immediately above regarding undetected failures noted in the log.

5. Extending Database Storage

If the database storage available to Oracle for the `ELIST_DATA`, `ELIST_INDEX`, `ETPFDD_DATA`, or `ETPFDD_INDEX` tablespaces becomes filled up during the processing performed by the **Run ELIST** or **Run ETEdit** features of the ELIST Software Segment, a message will be written to the terminal window that remains open throughout that processing. When this happens, a DBA will have to "extend" the database storage. This section explains how to do that.

Determine which tablespace filled up by examining the error messages, and then exit the feature that was being used.

A DBA should log in to the DBA account mentioned previously, invoke the **Extend Storage** feature of the **DBAdmR** application, and follow the instructions in the *System Administrator's Manual (SAM) for Database Administrator Runtime (DBAdmR)* for the use of that feature. Using that GUI-based tool, select the datastore(s) to be extended and specify the desired amount of additional space (expressed in MB). Before *that* can succeed, it may be necessary to use the **Identify Storage** feature of the **DBAdmR** application to identify additional storage (by partition and size) for **Extend Storage** to use. This use of **Identify Storage** is analogous to the use that was made of it when the ELIST Database Segment was installed (review those instructions in the *ELIST IP*).

This page intentionally left blank.

6. Notes

The following acronyms are (or may be) used in this document.

Acronym	Definition
API	Application Program Interface
APIRM	Application Program Interface Reference Manual
CD	Compact Disk
CM	Configuration Management
COE	Common Operating Environment
COTS	Commercial Off-the-Shelf
DB	Database
DBA	Database Administrator
DBAdmR	Database Administrator Runtime (DII COE segment prefix)
DBAdmS	Database Administrator Server (DII COE segment prefix)
DBO	Database Owner
DII	Defense Information Infrastructure
DISA	Defense Information Systems Agency
ELIST	Enhanced Logistics Intratheater Support Tool (DII COE segment prefix)
ETEdit	ETPFDD Editor
ETPFDD	Expanded Time-Phased Force Deployment Data
GB	Gigabyte(s)
GUI	Graphical User Interface
IP	Installation Procedures
I&RTS	Integration and Runtime Specification (DII COE document)
KB	Kilobyte(s)
LAN	Local Area Network
MB	Megabyte(s)
N/A	Not Applicable
NFS	Network File System
NT	New Technology (an Operating System for Microsoft Windows)
ORAS	ORACLE RDBMS (DII COE segment prefix)
PC	Personal Computer
RDBMS	Relational Database Management System
SAM	System Administrator's Manual
SQL	Structured Query Language

This page intentionally left blank.

7. Acknowledgements

Argonne National Laboratory is a Federally Funded Research and Development Center operated by The University of Chicago under contract W-31-109-ENG for the United States Department of Energy.

The development of ELIST and the preparation of this document were supported by funding from the Military Traffic Management Command Transportation Engineering Agency of the United States Army.

This page intentionally left blank

8. Documentation Improvement and Feedback

Comments and other feedback on this document should be directed to:

Phone: (630) 252-7217
Fax: (630) 252-5128
Email: dritz@anl.gov

This page intentionally left blank.